



**«Московский государственный технический
университет имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)**

Факультет «Информатика и системы управления»
Кафедра «Программное обеспечение ЭВМ и информационные технологии»

**Отчет по лабораторной работе №2
по курсу:
«Функциональное и логическое программирование»**

Студент группы ИУ7-64Б: Керимов А. Ш.
Преподаватели: Толпинская Н. Б.,
Строганов Ю. В.

Практическая часть

Задание 1. Что будет в результате вычисления выражений?

1. `(CAADR '((blue cude) (red pyramid))) ; red`
2. `(CDAR '((abc) (def) (ghi))) ; Nil`
3. `(CADR '((abc) (def) (ghi))) ; (def)`
4. `(CADDR '((abc) (def) (ghi))) ; (ghi)`

Задание 2. Напишите результат вычисления выражений:

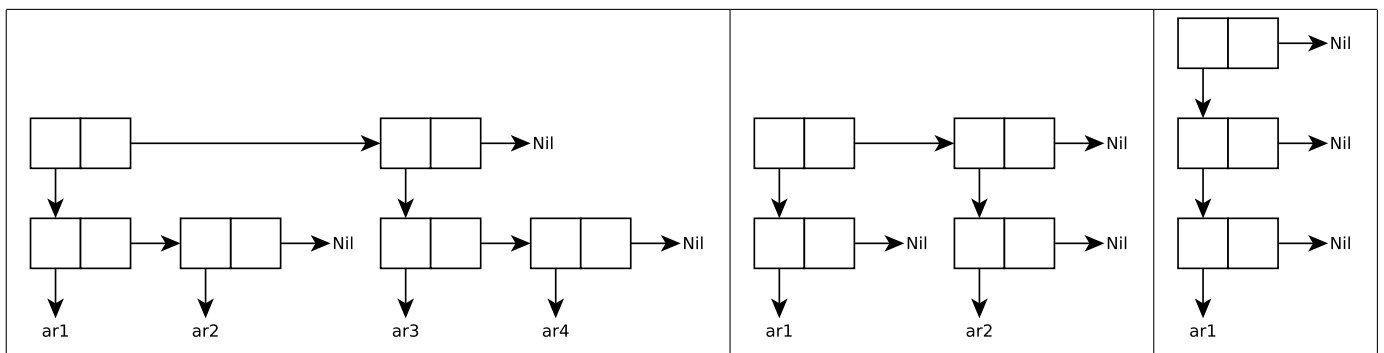
1. `(list 'Fred 'and Wilma)`
; The variable Wilma is unbound.
2. `(list 'Fred '(and Wilma))`
; (Fred (and Wilma))
3. `(cons Nil Nil)`
; (Nil)
4. `(cons T Nil)`
; (T)
5. `(cons Nil T)`
; (Nil . T)
6. `(list Nil)`
; (Nil)
7. `(cons (T) Nil)`
; The function T is undefined.
8. `(list '(one two) '(free temp))`
; ((one two) (free temp))
9. `(cons 'Fred '(and Wilma))`
; (Fred and Wilma)
10. `(cons 'Fred 'Wilma)`
; (Fred . Wilma)
11. `(list Nil Nil)`
; (Nil Nil)
12. `(list T Nil)`
; (T Nil)
13. `(list Nil T)`
; (Nil T)
14. `(cons T (list Nil))`
; (T Nil)
15. `(list (T) Nil)`
; The function T is undefined.
16. `(cons '(one two) '(free temp))`
; ((one two) free temp)

Задание 3. Написать функцию `(f ar1 ar2 ar3 ar4)`, возвращающую список: `((ar1 ar2) (ar3 ar4))`. Написать функцию `(f ar1 ar2)`, возвращающую список: `((ar1) (ar2))`. Написать функцию `(f ar1)`, возвращающую список: `((ar1))`. Представить результаты в виде списочных ячеек.

```
(defun f (a1 a2 a3 a4)
  '((,a1 ,a2) (,a3 ,a4))
)

(defun f (a1 a2)
  '((,a1) (,a2))
)

(defun f (a1)
  '(((,a1)))
)
```



Теоретическая часть

Базис Lisp

Базис Лиспа предельно лаконичен — атомы и структуры из простейших бинарных узлов плюс несколько базовых функций и функционалов. Базис содержит встроенные (примитивные) функции, которые анализируют, строят и разбирают любые структурные значения (`atom`, `eq`, `cons`, `car`, `cdr`), и встроенные специальные функции и функционалы, которые управляют обработкой структур, представляющих вычисляемые выражения (`quote`, `cond`, `lambda`, `eval`).

Классификация функций

Классификация функций:

1. чистые математические функции (имеют фиксированное количество аргументов и один результат);
2. формы (имеют произвольное количество аргументов или эти аргументы обрабатываются не все одинаково);
3. функциональные (в качестве одного из аргументов принимают описание функции).

Классификация базисных функций:

1. функции-селекторы: `car`, `cdr`;
2. функции-конструкторы: `cons`, `list`;
3. функции-предикаты: `atom`, `null`, `listp`, `consp`;
4. функции сравнения: `eq`, `eq1`, `equal`, `equalp`.

Как представляются списки в ОП?

Любая непустая структура Lisp в памяти представляется списковой ячейкой, хранящей два указателя: на голову (первый элемент) и хвост — всё остальное.

Функции `car` и `cdr` в разных случаях

Функция `car` обеспечивает доступ к первому элементу списка — его «голове», а функция `cdr` — к укороченному на один элемент списку — его «хвосту», т. е. к тому, что остается после удаления головы.

- | | |
|---|---|
| 1. <code>(car (A B C))</code>
; A | 5. <code>(cdr (A))</code>
; Nil |
| 2. <code>(car (A (B C)))</code>
; A | 6. <code>(cdr (A B C D))</code>
; (B C D) |
| 3. <code>(car ((A B) C))</code>
; (A B) | 7. <code>(cdr (A (B C)))</code>
; ((A B)) |
| 4. <code>(car A)</code>
; The variable A is unbound. | 8. <code>(cdr A)</code>
; The variable A is unbound. |